MeshLLM: Empowering Large Language Models to Progressively Understand and Generate 3D Mesh

Shuangkang Fang[1], I-Chao Shen[2], Yufeng Wang[1], Yi-Hsuan Tsai[3], Yi Yang[4], Shuchang Zhou[4], Wenrui Ding[1], Takeo Igarashi[2], Ming-Hsuan Yang[5],
[1]Beihang University [2]The University of Tokyo [3]Atmanity Inc [4]StepFun [5]UC Merced

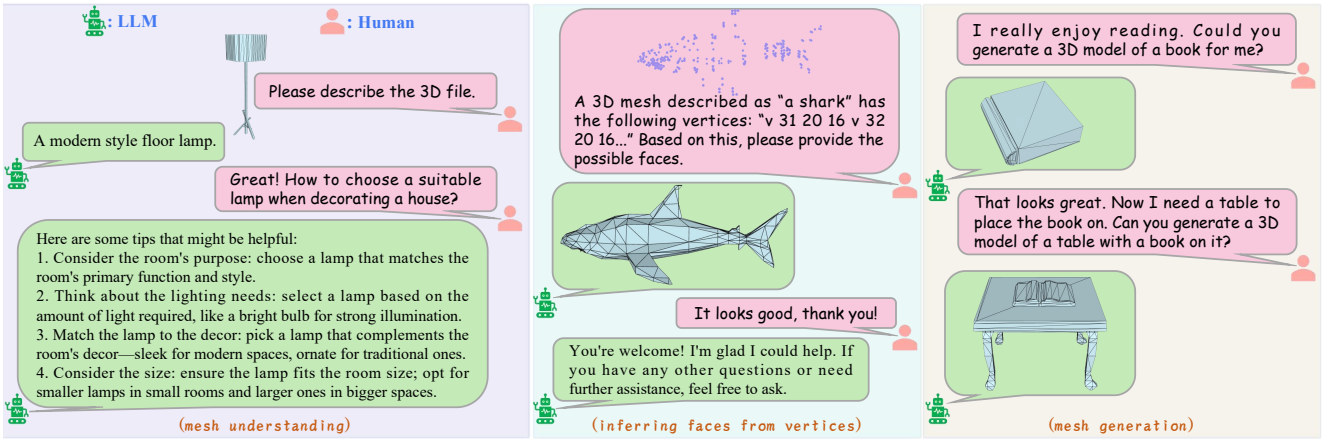


Figure 1. We propose MeshLLM, a method for effectively injecting text-serialized meshes into large language models, enabling the understanding and generation of 3D mesh through more natural conversational interactions.

Abstract

We present MeshLLM, a novel framework that leverages large language models (LLMs) to understand and generate text-serialized 3D meshes. Our approach addresses key limitations in existing methods, including the limited dataset scale when catering to LLMs' token length and the loss of 3D structural information during mesh serialization. We introduce a Primitive-Mesh decomposition strategy, which divides 3D meshes into structurally meaningful subunits. This enables the creation of a large-scale dataset with 1500k+ samples, almost 50× larger than previous methods, which aligns better with the LLM scaling law principles. Furthermore, we propose inferring face connectivity from vertices and local mesh assembly training strategies, significantly enhancing the LLMs' ability to capture mesh topology and spatial structures. Experiments show that MeshLLM outperforms the state-of-the-art LLaMA-Mesh in both mesh generation quality and shape understanding, highlighting its great potential in processing text-serialized 3D meshes.

1. Introduction

In recent years, large language models (LLMs), exemplified by the GPT [1, 6, 50, 53] series, have achieved groundbreaking advancements. Their powerful text generation and comprehension capabilities, along with their broad applicability, have continuously propelled them toward the goal of artificial general intelligence [3, 4, 13, 33, 58–61, 66]. Concurrently, the rise of multimodal learning has made the integration of language models with other modalities a prominent research focus, such as vision and speech [23, 44–46, 56, 71]. However, the modeling and comprehension of 3D data by LLMs remain underexplored. With the rapid development of virtual reality and robotic interaction, equipping LLMs with 3D perception and spatial reasoning capabilities has become a pressing challenge.

Against this backdrop, existing research has attempted to integrate LLMs with 3D data [11, 22, 27, 34, 67, 69]. These methods typically rely on pretrained 3D encoders to map 3D structures into discrete token sequences before inputting them into LLMs for reasoning and question-answering tasks. While these methods have demonstrated the feasibility of LLMs in 3D tasks, they face several chal-

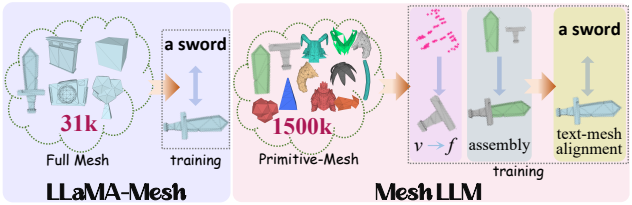


Figure 2. **Differences between LLaMA-Mesh and MeshLLM**. LLaMA-Mesh applies a single text-mesh alignment optimization strategy on only 31k available meshes. In contrast, our proposed MeshLLM leverages local mesh patches and thus expands the trainable data to 1500k, enhancing model performance through vertex-to-face prediction and local mesh assembly strategies.

lenges, including dependency on specific encoders, the need to expand the LLMs' vocabulary, and potential information loss during encoding. Recently, LLaMA-Mesh [63] has proposed representing 3D meshes directly in text-serialized format, enabling LLMs to parse and generate 3D meshes in their native text-processing manner. This approach is based on two key observations. First, text-based mesh serialization requires no additional encoders, making it naturally compatible with LLMs' text modeling capabilities. Second, LLMs show compatibility with other rule-based text formats such as programming code and SVG, suggesting potential for LLM-based approaches using the simpler mesh.

Despite pioneering the exploration of understanding and generating text-serialized mesh, LLaMA-Mesh poses new challenges to the research community: 1) *Data scale limitations*: As suggested by the Scaling Law [37], large-scale data is key to enhancing LLMs performance. However, due to the limitation of LLMs' token length, LLaMA-Mesh discards a large number of long mesh sequences, and only 31k samples are used for training, significantly constraining its potential. 2) *Insufficient 3D structural awareness*: Directly learning text-serialized mesh representations causes LLMs to overlook the inherent spatial structure (e.g., connectivity and semantic segmentation) of 3D mesh. Introducing dedicated mechanisms within LLMs to capture and maintain structural information remains a technical bottleneck.

To address these challenges, this paper proposes MeshLLM, which differs from LLaMA-Mesh as shown in Fig. 2, and includes the following key designs: 1) *Primitive-Mesh construction*: As shown in Fig. 3, we first utilize K-Nearest Neighbors (KNN) to pre-decompose complex meshes into multiple localized subunits—termed Primitive-Mesh. This simple approach enables us to quickly construct a large-scale dataset comprising 1500k+ training samples. However, these samples may not capture the underlying semantic coherence. To this end, we further leverage a high-quality 3D mesh segmentation tool [68] to curate a subset of over 100K semantic-level Primitive-Mesh samples. These samples provide accurate structural and semantic details, much like LLMs process local windows in natural language, further enhancing the model's comprehension of 3D mesh. 2) *Task-Specific training strategies*: Based on the constructed dataset, we design two additional training tasks of vertex-face prediction and local mesh assembly. The former enhances LLMs' topological reasoning abilities by inferring face connectivity from vertices, while the latter improves global modeling capabilities by reconstructing complete meshes from local structures. This strategy enables the LLMs to robustly capture both local and global 3D structural information from the text-serialized mesh. Complementing these tasks, we implement a progressive training process that transitions from large-scale pretraining on the extensive Primitive-Mesh dataset to targeted fine-tuning on specific tasks.

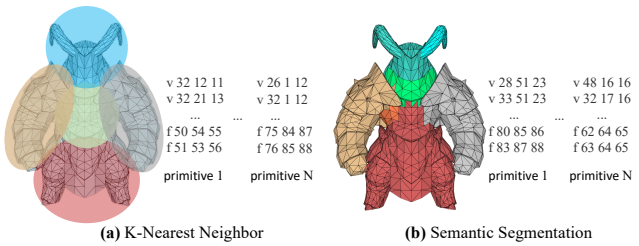
The main contributions of our work are as follows:

- We introduce a mesh decomposition strategy to create 1500k+ Primitive-Meshes, expanding the scale of the trainable dataset by nearly 50 times, which deeply enhances the LLMs performance in text-serialized mesh generation and understanding.
- We propose the MeshLLM framework, incorporating vertex-face prediction and local mesh assembly training tasks to enhance LLMs' structural awareness of 3D mesh.
- Experimental results show that MeshLLM significantly outperforms LLaMA-Mesh, offering new insights into integrating LLMs with text-serialized mesh representations.

2. Related Work

Large Language Models and Multimodal Expansion. In recent years, LLMs have achieved remarkable advancements in natural language processing. Models such as GPT [1, 6, 50, 53], LLaMA [25, 60, 61], and the DeepSeek [5, 26, 42, 43] series have demonstrated exceptional capabilities in text generation and comprehension, driven by massive datasets and large-scale parameterization. LLMs are progressively expanding into the multimodal domain [21, 24, 29, 39, 40, 44, 45, 70, 71]. Approaches like LLaVA [44, 45], Video-LLaMA [71] and SpeechGPT [70] extend LLMs to image, video and speech processing, showcasing their potential in temporal and visual understanding. Due to challenges such as the difficulty of directly textualizing image and speech data or the excessive length of textualized data, these methods often require customized tokenizers to embed multimodal data into a unified space, thereby bridging the gap with language.

Mesh Generation. Meshes are a fundamental 3D representation widely used in computer vision and graphics. Various methods have been proposed for mesh generation. Some approaches extract meshes from other 3D representations [12, 14, 19, 20, 28, 35, 36, 48], such as SDF [52],



(a) K-Nearest Neighbor **(b)** Semantic Segmentation

Figure 3. **Illustration of Primitive-Mesh**. We utilize both KNN clustering and semantic segmentation to partition meshes into Primitive-Meshes that retain local structural information. This strategy greatly expands the scale of the trainable dataset.

NeRF [47], and Gaussian Spaltting [38], prioritizing shape and color accuracy. However, the resulting meshes are often overly dense [9, 10]. In contrast, this work focuses on direct mesh generation. Existing methods in this way typically convert meshes into sequential representations [2, 8–10, 32, 49, 57, 64, 65] using predefined sorting strategies and learning their distribution on large-scale datasets for generation. For example, PolyGen [49] employs an autoregressive Transformer to generate mesh vertices and faces sequentially. MeshGPT [57] adopts a similar framework but encodes mesh using a pretrained VQ-VAE. MeshXL [8] combines implicit neural embeddings with explicit coordinates for generation. MeshAnything [9] converts 3D representations into point clouds before generating meshes, with MeshAnythingV2 [10] introducing a more compact sequence representation for improved efficiency.

Mesh Understanding. Research on mesh understanding remains relatively limited. Y2Seq2Seq [30] aggregates multi-view semantic information to enhance mesh comprehension, while ShapeCaptioner [31] introduces part detection for finer-grained descriptions, and ShapeGPT [69] maps 3D shapes to word embeddings using an encoder to capture semantic information. Despite these advancements, existing methods treat mesh generation and understanding as separate tasks. Recently, LLaMA-Mesh [63] has explored integrating textualized meshes with LLMs, achieving unified generation and understanding, highlighting the potential of LLMs in directly modeling text-serialized meshes. However, LLaMA-Mesh suffers from inefficient utilization of existing datasets and loss of structural information during text serialization. This paper proposes improvements to address these issues, aiming to achieve more accurate mesh generation and understanding using LLMs.

3. Method

We first describe the process of converting 3D mesh data into a textual sequence compatible with LLMs. Next, we introduce the concept of Primitive-Mesh. Finally, we outline the supervised fine-tuning tasks designed for LLMs, along with corresponding data formats and training workflows.

3.1. Preliminaries: Text-serialized Mesh

To enable 3D mesh data to be directly modeled by LLMs, we need to convert it into purely textual sequences. Similar to LLaMA-Mesh [63], we adopt the OBJ-format as the fundamental representation for a mesh. Given a mesh $\mathcal{M} = (\mathcal{V}, \mathcal{F})$, where $\mathcal{V} = \{v_i\}_{i=1}^{N_v}$ represents N_v vertices, each vertex $v_i \in \mathbb{R}^3$ corresponds to a spatial coordinate, i.e., $v_i = (x_i, y_i, z_i)$. The set of faces $\mathcal{F} = \{f_j\}_{j=1}^{N_f}$ consists of N_f triangular face elements defined by three vertex indices.

The mesh is textualized through the following steps: 1) *Quantization*: The coordinate values of the mesh vertices are mapped to the integer values in $[0, 64]$, thereby confining infinite numerical values to a finite range. Since characters '*v*', '*f*', and digits 0 through 64 are common symbols, there is no need to modify the LLMs' tokenizer or vocabulary. 2) *Sorting*: Employing a sorting strategy akin to PolyGen [49], we assign a unique sequence to each mesh. Specifically, vertices are sorted in ascending order based on their z-y-x coordinates. Faces are then sorted according to the smallest vertex index within each face. 3) *Textual sequence unfolding*: The sorted mesh is flattened into a text format, with special characters (e.g., newline symbols) replaced to yield a final textual sequence representation:

$$\mathcal{M} = [\text{Vertex List}] \| [\text{Face List}], \tag{1}$$

where $\|$ denotes sequence concatenation.

3.2. Primitive-Mesh

Training LLMs directly on the above text-serialized mesh poses several challenges: 1) The limited token length of LLMs constrains the number of trainable samples; 2) The text sequences fail to convey the intrinsic 3D structure; 3) Learning from long sequences is inherently difficult for LLMs. To address these issues, we propose decomposing a mesh into multiple localized components, termed Primitive-Mesh, which can be formulated as follows:

$$\mathcal{M} = \{\mathcal{M}_1, \mathcal{M}_2, \ldots, \mathcal{M}_N\}, \tag{2}$$

where N denotes the number of Primitive-Mesh units. This design is motivated by the observation that LLMs benefit from truncated local text in natural language tasks. Similarly, localized mesh components retain spatial information, aiding LLMs in perceiving 3D spatial structures. Furthermore, Primitive-Mesh sequences are shorter and significantly more numerous, allowing for deeper exploitation of LLMs' potential on large-scale datasets.

As shown in Fig. 3, we construct Primitive-Mesh using two strategies: 1) *KNN-Based*: Given a mesh $\mathcal{M}$, we begin by densely sampling point clouds from the mesh and then apply farthest point sampling (FPS) and KNN to identify central points and point clusters, thereby partitioning

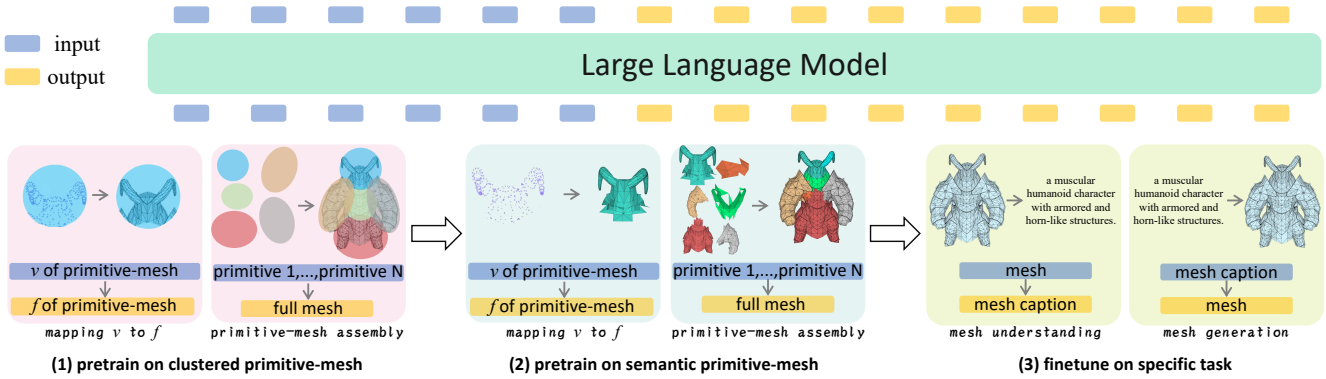


Figure 4. **Illustration of the MeshLLM framework**. We adopt a progressive training process: *Stage* 1: Training on Primitive-Meshes obtained through KNN clustering, where two tasks are performed: predicting faces from vertices and assembling complete meshes from Primitive-Meshes. *Stage* 2: Training on more refined Primitive-Meshes generated by semantic segmentation, performing the same tasks as in Stage 1. *Stage* 3: Training on tasks specific to mesh generation and understanding.

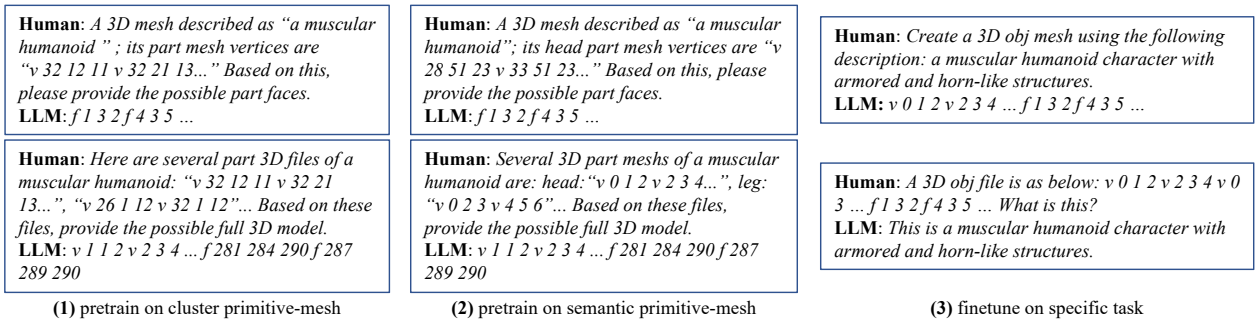


Figure 5. **Example of the constructed SFT data for training LLM**.

the mesh into multiple local regions. This approach is computationally efficient and applicable to any existing mesh dataset. Using this strategy, we generate over 1500k+ training samples. However, the mesh parts obtained through KNN may lack semantic coherence. Therefore, we construct an additional dataset to complement and refine the existing data. 2) *Semantic-based*: To obtain Primitive-Meshes with well-defined semantic boundaries, we leverage 3DSAMPart [68] to perform mesh segmentation on a curated subset after aesthetically filtered [55]. This yields over 100k+ high-quality Primitive-Mesh samples. As depicted in Fig. 3, this method accurately segments a humanoid mesh into regions such as the head, hands, and legs. These semantically meaningful datasets further enhance the LLMs' comprehension of high-level semantic information.

3.3. Training Task Design

Building on the constructed dataset, we design four supervised fine-tuning tasks to enhance the LLMs' ability to understand and generate 3D meshes, as shown in Fig. 4.

Vertex-Face Prediction. Given a set of vertex coordinates $\mathcal{V}$ and its corresponding faces $\mathcal{F}$, the LLM is optimized according to the following objective:

$$\max_{\theta} P(\mathcal{F} \mid \mathcal{V}, \theta), \tag{3}$$

where P is modeled by the LLM, and θ is its parameters. This task enables the LLM to predict face connectivity given vertices, thereby learning the topological relationships between vertices.

Mesh Assembly. Given a complete mesh $\mathcal{M}$ and its corresponding set of Primitive-Mesh components $\{M_i\}_{i=1}^{k}$, the LLM learns to reconstruct the full mesh by optimizing:

$$\max_{\theta} P(\mathcal{M} \mid \{\mathcal{M}_i\}_{i=1}^{k}, \theta). \tag{4}$$

This task captures the geometric relationships between local Primitive-Mesh units, mitigating the loss of 3D spatial information inherent in textual serialization, thereby improving the model's ability to infer mesh structures.

Mesh Understanding. Given a mesh $\mathcal{M}$ and its textual description $\mathcal{T}$, the following learning objective is constructed:

$$\max_{\theta} P(\mathcal{T} \mid \mathcal{M}, \theta). \tag{5}$$

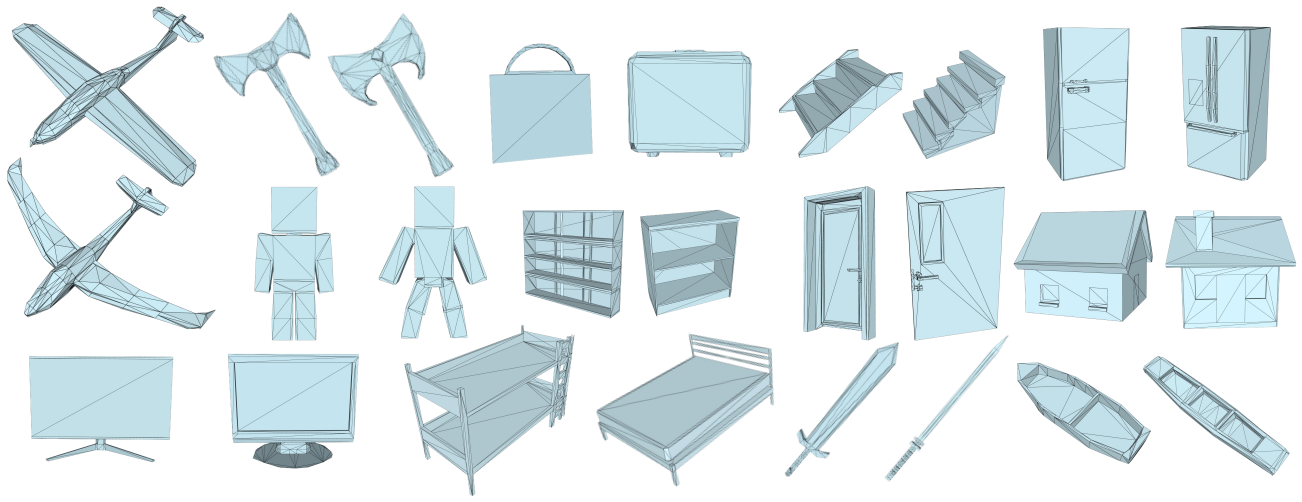


Figure 6. **Gallery results**. MeshLLM demonstrates an ability to generate diverse and high-quality meshes.

This enables the LLM to generate accurate and fluent descriptions based on mesh data, thereby acquiring an understanding of high-level semantic information.

Mesh Generation. Given a textual description $\mathcal{T}$ and a Mesh $\mathcal{M}$, the LLM is trained to optimize:

$$\max_{\theta} P(\mathcal{M} \mid \mathcal{T}, \theta). \tag{6}$$

This encourages LLM to learn to generate plausible mesh structures from textual descriptions.

It is important to note that these tasks are not independent but are implemented in a progressive training process as shown in Fig. 4. Initially, vertex-face prediction and mesh assembly tasks familiarize the model with fundamental mesh structures and local semantics. Subsequently, mesh understanding and mesh generation tasks refine the model's grasp of complex 3D structures and high-level semantics.

3.4. SFT Data Curation

In LLMs' downstream task alignment training, Supervised Fine-Tuning (SFT) is one of the most widely used strategies. As illustrated in Fig. 5, we construct various forms of SFT data encompassing the aforementioned four training tasks. It employs high-quality input-output data pairs with standard language modeling objectives to fine-tune LLMs, thereby better adapting LLM to 3D tasks.

4. Experimental Results

4.1. Implementation Details

Dataset. MeshLLM is trained primarily on two datasets: Objaverse-XL [15] (including the sketch and GitHub subsets) and ShapeNet [7]. For the construction of KNN-based Primitive-Mesh data, we perform clustering with 2 to 10 categories based on the number of faces in the original mesh, resulting in over 1500k+ Primitive-Meshes. For the creation of semantic-based Primitive-Meshes, we first conduct an aesthetic evaluation [55] of the original datasets, marking approximately 25k+ high-quality mesh subsets. We then generate semantic-level Primitive-Meshes on this subset using the SamPart3D method [68], yielding over 100k+ semantic-level Primitive-Meshes. This process utilized 128 A800 GPUs and took approximately 3 days. For the mesh assembly, understanding and generation tasks involving full-mesh data, we follow a procedure similar to PolyGen [49]. Specifically, we apply planar simplification to meshes with fewer than 3k faces and restrict the number of faces in the final full-mesh representation to 800, ensuring compatibility with the LLM's maximum token length. We follow dataset split configurations from previous works [8, 49], extracting 10% of the 4 subsets (chair, table, bench, lamp) from ShapeNet and 1K samples from Objaverse-XL as the test set to evaluate the quality of mesh generation and understanding, respectively.

Training Details. We use LLaMA-8B-Instruct [25] as the base LLM model and finetune its full 8 billion parameters based on our constructed data. We employ the AdamW optimizer with a learning rate of 2e-5 and set the maximum context length to 8192. We train for 2 epochs on the KNN-based Primitive-Mesh dataset, 3 epochs on the semantic Primitive-Mesh dataset, and 3 epochs on the mesh-generation and understanding datasets. Additionally, to mitigate catastrophic forgetting and retain the LLM's conversational capabilities, we randomly sample the data from the previous phase and ultra-chat dataset [17] with a 30% probability during each training phase. We employ data augmentation when training meshes, including random scaling

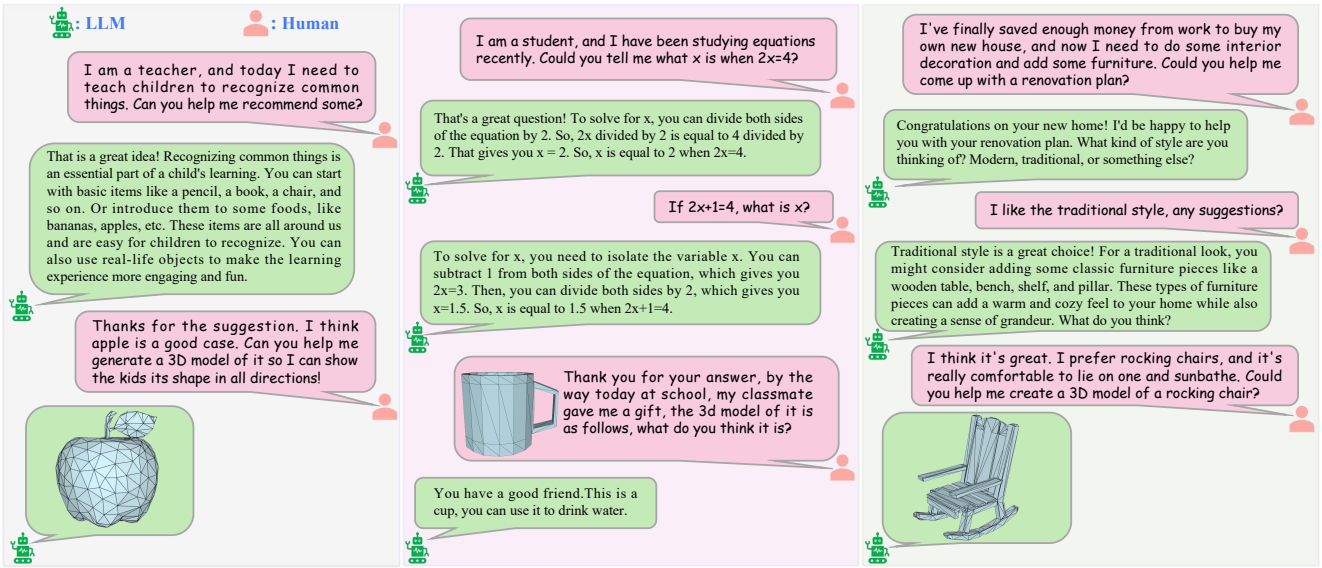


Figure 7. **Dialogue results**. MeshLLM extends the capabilities of LLMs to the domain of 3D mesh while retaining their advanced dialogue abilities, such as question-answering and mathematical reasoning. This expansion enables MeshLLM to understand and generate 3D meshes through natural and intuitive language interactions, further solidifying LLMs as versatile and powerful tools.

and random translation. The training process is conducted using 128 A800 GPUs and took approximately 6 days.

Metrics. To evaluate the quality of mesh generation, we adopt the same metrics as previous studies [8, 18, 49, 57]. These metrics include Minimum Matching Distance (MMD, lower is better), Coverage (COV, higher is better), and 1-Nearest Neighbor Accuracy (1-NNA, the optimal value is 50%). Please refer to the supplementary materials for a detailed explanation. We also calculate the Frechet Inception Distance (FID) and Kernel Inception Distance (KID) on 8 rendered images for feature-level evaluation. We generate 1000 meshes for each evaluated category and report their average metrics. For the mesh understanding task, we use the BLEU-1 [51], CIDEr [62], METEOR [16], and ROUGE [41] metrics to evaluate the accuracy of the generated captions. In addition, we render 8 different images of the meshes and compute the CLIP similarity [54] between these images and the text to assess the alignment between the mesh and the text.

Baselines. Our method primarily focuses on enabling LLMs to perceive and generate text-serialized mesh. The most directly related baseline to our approach is LLaMA-Mesh [63]. Additionally, we also compare the mesh quality with MeshXL (under text-conditional settings) [8] and PolyGen (class-conditional) [49]. All comparative experiments used the same text prompt, except for PolyGen, which only supports inputting the category of mesh.

4.2. Dialogue Ability

We design a variety of interactive dialogue scenarios that simulate user interactions with mesh through natural language instructions. Experimental results in Fig. 1 and Fig. 7 indicate that MeshLLM not only generates 3D mesh structures that faithfully match the textual descriptions provided by the user but also provides explanatory feedback on mesh details and topological structures during the dialogue while retaining its inherent natural language generation ability, facilitating smooth and coherent multi-turn conversations. These findings demonstrate that our approach successfully integrates text-serialized 3D information into LLMs. In particular, the constructed data sets and training pipeline are fully compatible with any existing LLM without necessitating additional complex encoder-decoder designs.

4.3. Performance Evaluation

We compare MeshLLM with existing methods across two primary dimensions: mesh generation quality and mesh understanding capability.

Mesh Generation. Fig. 6 demonstrates the ability of our method to generate diverse meshes. We further compare it with state-of-the-art methods in Fig. 8. It can be seen that MeshLLM generates finely detailed geometric structures, achieving significantly superior results compared to LLaMA-Mesh. Moreover, when compared with methods specifically designed for mesh generation like PolyGen and MeshXL, the overall performance of MeshLLM is comparable. Quantitative evaluations, as presented in Tab. 1,

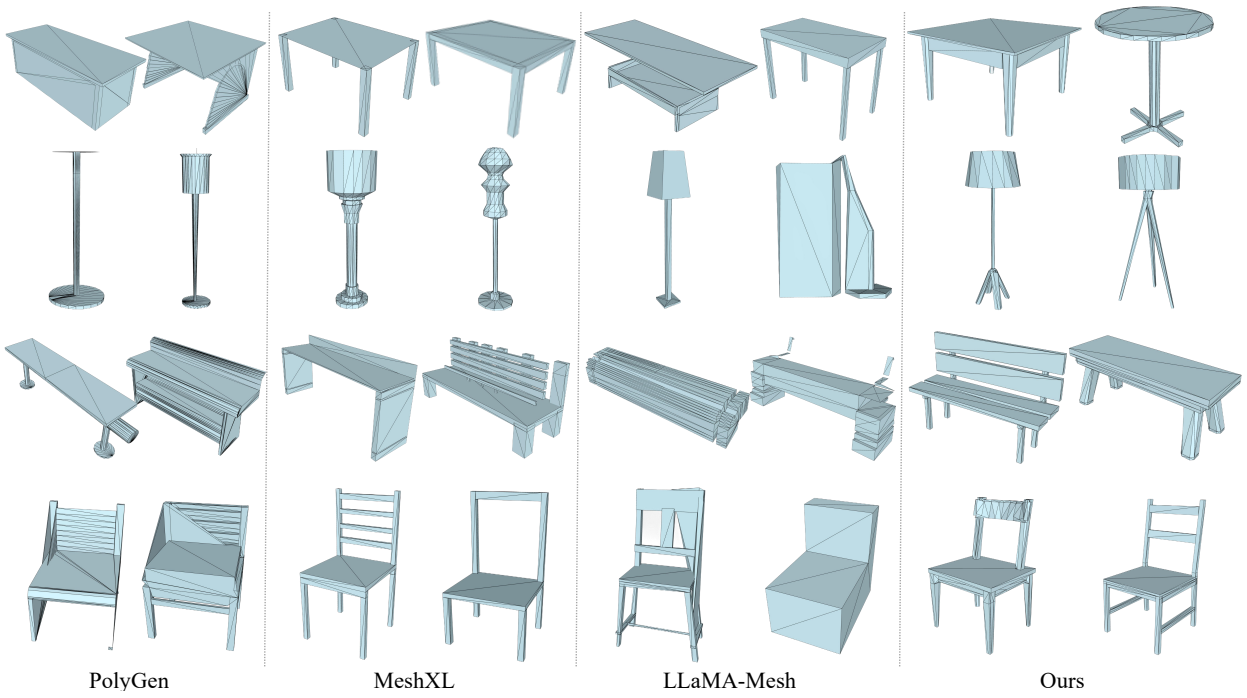


Figure 8. **Comparisons on the mesh generation**. MeshLLM generates 3D meshes with clean geometric details, outperforming the LLM-based LLaMA-Mesh and achieving performance comparable to Polygen and MeshXL, which are specifically designed for mesh generation.

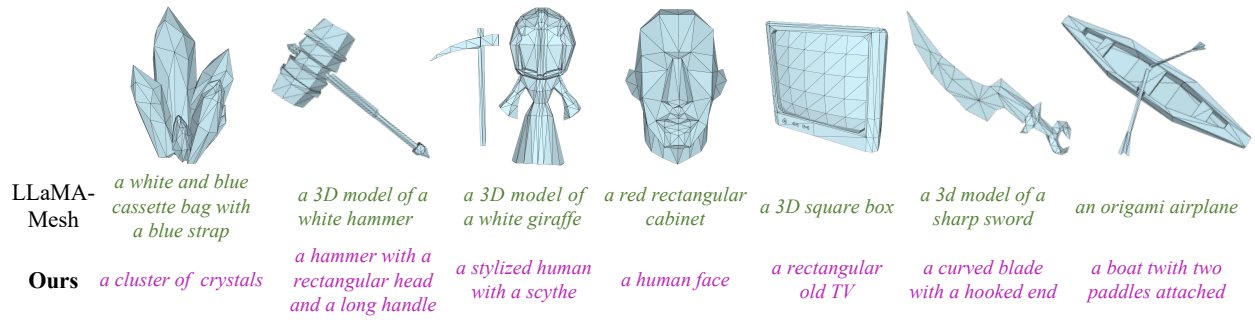


Figure 9. **Comparisons on the mesh understanding**. The green text is generated by LLaMA-Mesh, while the purple text is produced by our MeshLLM. MeshLLM better captures the semantic information of the mesh, generating more accurate textual descriptions.

reveal that our method surpasses LLaMA-Mesh on multiple metrics and achieves a performance comparable to that of MeshXL, thereby validating the effectiveness of our Primitive-Mesh construction strategy and training task design. It is worth noting that while MeshXL and PolyGen excel in mesh generation tasks, neither possesses mesh understanding or interactive dialogue capabilities, which are unique advantages of our LLM-based approach.

Mesh Understanding. As shown in Fig. 9 and Tab. 2, we qualitatively and quantitatively evaluate the mesh textual descriptions generated by the LLM. MeshLLM excels at capturing high-level semantic information of meshes. The generated descriptions are fluent and accurate and effectively reflect the structural characteristics of the meshes, which significantly surpass the LLaMA-Mesh baseline. The improvement primarily stems from the finer-grained semantic information embedded in Primitive-Meshes, as well as the mesh assembly task, which reinforces the connection between local and global semantics.

4.4. Ablation Studies

We conduct a series of ablation experiments, the results of which are summarized in Tab. 3. The primary ablation settings include: 1) *KNN-based Primitive-Mesh*: This design

Table 1. **Quantitative comparisons of mesh quality**. Bold and underline denote the 1st and 2nd best-performing models, respectively. MeshLLM surpasses the same-type method LLaMA-Mesh and is comparable to encoder-based MeshXL. However, unlike our MeshLLM, MeshXL lacks the ability to understand mesh and dialogue with users.

	Capacities			Chair					Table					Bench					Lamp				
Method	generation	understanding	dialogue	COV↑	MMD↓	1-NNA	FID↓	KID↓	COV↑	MMD↓	1-NNA	FID↓	KID↓	COV↑	MMD↓	1-NNA	FID↓	KID↓	COV↑	MMD↓	1-NNA	FID↓	KID↓
PolyGen [49]	√	×	×	8.23	15.72	88.44	60.20	41.92	42.92	3.94	70.38	56.17	15.72	32.50	4.46	88.75	69.93	12.35	36.22	7.48	75.04	63.89	10.47
MeshXL [8]	√	×	×	46.16	**3.45**	**58.76**	**40.33**	2.41	47.74	3.18	**56.74**	42.64	**1.42**	**51.42**	**2.05**	**41.42**	41.8	1.26	**51.04**	4.72	**46.13**	**33.55**	**1.06**
LLaMA-Mesh [63]	√	√	√	19.53	8.64	77.78	49.83	23.37	38.98	4.72	75.60	60.49	10.71	36.57	4.22	70.53	56.74	8.95	31.06	9.98	82.76	65.29	12.01
MeshLLM	√	√	√	**47.33**	5.72	60.82	42.39	**2.25**	**49.26**	**3.15**	58.77	**39.59**	4.26	49.38	3.34	60.79	**36.63**	**1.09**	49.85	**3.30**	59.48	35.70	1.44

Table 2. **Quantitative comparisons of mesh understanding**. MeshLLM significantly surpasses the LLaMA-Mesh method.

Method	BLEU-1↑	CIDEr↑	Meteor↑	ROUGE↑	CLIP↑
LLaMA-Mesh	0.483	0.397	0.194	0.356	0.124
MeshLLM	**0.763**	**1.753**	**0.445**	**0.702**	**0.391**

Table 3. **Ablation studies of MeshLLM**. "PM" denotes Primitive-Mesh. We report the impact of key components on mesh generation (chair class) and understanding.

	Mesh Generation				
	COV↑	MMD↓	1-NNA	FID↓	KID↓
w/o KNN PM	42.36	5.74	72.40	49.33	6.44
w/o semantic PM	41.36	6.06	68.87	52.76	8.29
w/o v to f	44.80	5.81	61.77	48.68	4.90
w/o mesh assembly	40.17	6.43	70.25	54.26	7.32
Full	**47.33**	**5.72**	**60.82**	**42.39**	**2.25**
	Mesh Understanding				
	BLEU-1↑	CIDEr↑	Meteor↑	ROUGE↑	CLIP↑
w/o KNN PM	0.692	0.921	0.357	0.610	0.324
w/o semantic PM	0.646	0.782	0.301	0.543	0.282
w/o v to f	0.737	1.229	0.433	0.627	0.376
w/o mesh assembly	0.705	0.894	0.359	0.596	0.344
Full	**0.763**	**1.753**	**0.445**	**0.702**	**0.391**

is critical for constructing a large-scale usable dataset. Removing it leads to a significant decline in all evaluation metrics, underscoring its essential role in the MeshLLM framework. 2) *Semantic-based Primitive-Mesh*: This component is derived from high-quality, filtered meshes, providing more accurate and rich semantic information. Excluding it results in a slight reduction in mesh generation quality and a marked degradation in mesh understanding performance. 3) *Vertex-Face prediction strategy*: This module facilitates the learning of topological relationships between vertices and faces. Its removal causes deviations in reconstructing the mesh topology, resulting in a significant drop in overall generation quality. 4) *Mesh assembly strategy*: Designed to capture global spatial relationships among text-serialized 3D data, this module is crucial for enhancing global structural reconstruction. Ablating this component also leads to a pronounced performance decrease.

5. Limitation and Future Work

While MeshLLM shows the potential of LLMs for 3D mesh understanding and generation, certain limitations remain, highlighting future research areas: 1) The scale of available mesh data is still vastly smaller than the corpora used in NLP. It is critical to construct larger and higher-quality datasets to fully leverage LLMs' abilities. 2) The limited dataset size results in imprecise alignment between text and geometric structures, constraining the ability to perform fine-grained generation and refinement of meshes. Incorporating additional modalities, like images, to encode structural information could enhance LLMs' performance, especially when data is scarce. 3) Handling more complex meshes can benefit from compact serialization methods (e.g., MeshAnything-V2 [10]) and LLMs with larger token capacities. These optimizations are orthogonal to our current work. 4) Another promising direction is designing external intelligent agents to analyze interaction results and leverage reinforcement learning to refine geometric accuracy and achieve specific aesthetic objectives.

6. Conclusions

In this paper, we propose MeshLLM, a novel approach that rethinks the paradigm of generating text-serialized meshes using Large Language Models, which addresses two key limitations of existing approaches: 1) insufficient utilization of available datasets and 2) disruption of underlying 3D structures caused by 2D serialization. Our solution introduces a Primitive-Mesh strategy to divide meshes, expanding the trainable dataset to over 1500k samples. Additionally, we construct a meticulously curated dataset containing more than 100k high-quality, semantically segmented meshes to enhance the LLM's ability to understand and reason mesh structures. Building on the constructed dataset, we propose a structured training paradigm that models meshes hierarchically from vertices to faces and mesh assembly, enabling LLMs to effectively perceive the 3D world. We hope that our findings will foster deeper integration between LLMs and the 3D mesh domain, offering the research community a new perspective for developing powerful multimodal intelligent agents.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (U24B6013), China Scholarship Council (202406020139), and Japan JSPS Grant-in-Aid (JP23K16921).

References

[1] Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023. 1, 2

[2] Antonio Alliegro, Yawar Siddiqui, Tatiana Tommasi, and Matthias Nießner. Polydiff: Generating 3d polygonal meshes with diffusion models. *arXiv preprint arXiv:2312.11417*, 2023. 3

[3] Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*, 2023. 1

[4] Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. Qwen technical report. *arXiv preprint arXiv:2309.16609*, 2023. 1

[5] Xiao Bi, Deli Chen, Guanting Chen, Shanhuang Chen, Damai Dai, Chengqi Deng, Honghui Ding, Kai Dong, Qiushi Du, Zhe Fu, et al. Deepseek llm: Scaling open-source language models with longtermism. *arXiv preprint arXiv:2401.02954*, 2024. 2

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33:1877–1901, 2020. 1, 2

[7] Angel X Chang, Thomas Funkhouser, Leonidas Guibas, Pat Hanrahan, Qixing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, et al. Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*, 2015. 5

[8] Sijin Chen, Xin Chen, Anqi Pang, Xianfang Zeng, Wei Cheng, Yijun Fu, Fukun Yin, Billzb Wang, Jingyi Yu, Gang Yu, et al. Meshxl: Neural coordinate field for generative 3d foundation models. *Advances in Neural Information Processing Systems*, 37:97141–97166, 2025. 3, 5, 6, 8

[9] Yiwen Chen, Tong He, Di Huang, Weicai Ye, Sijin Chen, Jiaxiang Tang, Xin Chen, Zhongang Cai, Lei Yang, Gang Yu, et al. Meshanything: Artist-created mesh generation with autoregressive transformers. *arXiv preprint arXiv:2406.10163*, 2024. 3

[10] Yiwen Chen, Yikai Wang, Yihao Luo, Zhengyi Wang, Zilong Chen, Jun Zhu, Chi Zhang, and Guosheng Lin. Meshanything v2: Artist-created mesh generation with adjacent mesh tokenization. *arXiv preprint arXiv:2408.02555*, 2024. 3, 8

[11] Yilun Chen, Shuai Yang, Haifeng Huang, Tai Wang, Runsen Xu, Ruiyuan Lyu, Dahua Lin, and Jiangmiao Pang. Grounded 3d-llm with referent tokens. *arXiv preprint arXiv:2405.10370*, 2024. 1

[12] Yen-Chi Cheng, Hsin-Ying Lee, Sergey Tulyakov, Alexander G Schwing, and Liang-Yan Gui. Sdfusion: Multimodal 3d shape completion, reconstruction, and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4456–4465, 2023. 2

[13] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. Palm: Scaling language modeling with pathways. *Journal of Machine Learning Research*, 24(240): 1–113, 2023. 1

[14] Angela Dai and Matthias Nießner. Scan2mesh: From unstructured range scans to 3d meshes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5574–5583, 2019. 2

[15] Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems*, 36:35799–35813, 2023. 5

[16] Michael Denkowski and Alon Lavie. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 376–380, 2014. 6

[17] Ning Ding, Yulin Chen, Bokai Xu, Yujia Qin, Zhi Zheng, Shengding Hu, Zhiyuan Liu, Maosong Sun, and Bowen Zhou. Enhancing chat language models by scaling high-quality instructional conversations. *arXiv preprint arXiv:2305.14233*, 2023. 5

[18] Ziya Erkoç, Fangchang Ma, Qi Shan, Matthias Nießner, and Angela Dai. Hyperdiffusion: Generating implicit neural fields with weight-space diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 14300–14310, 2023. 6

[19] Shuangkang Fang, Dacheng Qi, Weixin Xu, Yufeng Wang, Zehao Zhang, Xiaorong Zhang, Huayu Zhang, Zeqi Shao, and Wenrui Ding. Efficient implicit sdf and color reconstruction via shared feature field. In *Proceedings of the Asian Conference on Computer Vision*, pages 3499–3516, 2024. 2

[20] Shuangkang Fang, Dacheng Qi, Weixin Xu, Yufeng Wang, Zehao Zhang, Xiaorong Zhang, Huayu Zhang, Zeqi Shao, and Wenrui Ding. Efficient implicit sdf and color reconstruction via shared feature field. In *Proceedings of the Asian Conference on Computer Vision*, pages 3499–3516, 2024. 2

[21] Shuangkang Fang, Yufeng Wang, Yi-Hsuan Tsai, Yi Yang, Wenrui Ding, Shuchang Zhou, and Ming-Hsuan Yang. Chat-edit-3d: Interactive 3d scene editing via text prompts. In *European Conference on Computer Vision*, pages 199–216. Springer, 2024. 2

[22] Rao Fu, Jingyu Liu, Xilun Chen, Yixin Nie, and Wenhan Xiong. Scene-llm: Extending language model for 3d visual understanding and reasoning. *arXiv preprint arXiv:2403.11401*, 2024. 1

[23] Zhiqi Ge, Hongzhe Huang, Mingze Zhou, Juncheng Li, Guoming Wang, Siliang Tang, and Yueting Zhuang.

Worldgpt: Empowering llm as multimodal world model. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 7346–7355, 2024. 1

[24] Rohit Girdhar, Alaaeldin El-Nouby, Zhuang Liu, Mannat Singh, Kalyan Vasudev Alwala, Armand Joulin, and Ishan Misra. Imagebind: One embedding space to bind them all. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15180–15190, 2023. 2

[25] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407, 2024. 2, 5

[26] Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, et al. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025. 2

[27] Ziyu Guo, Renrui Zhang, Xiangyang Zhu, Yiwen Tang, Xianzheng Ma, Jiaming Han, Kexin Chen, Peng Gao, Xianzhi Li, Hongsheng Li, et al. Point-bind & point-llm: Aligning point cloud with multi-modality for 3d understanding, generation, and instruction following. *arXiv preprint arXiv:2309.00615*, 2023. 1

[28] Anchit Gupta, Wenhan Xiong, Yixin Nie, Ian Jones, and Barlas Oğuz. 3dgen: Triplane latent diffusion for textured mesh generation. *arXiv preprint arXiv:2303.05371*, 2023. 2

[29] Andrey Guzhov, Federico Raue, Jörn Hees, and Andreas Dengel. Audioclip: Extending clip to image, text and audio. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 976–980. IEEE, 2022. 2

[30] Zhizhong Han, Mingyang Shang, Xiyang Wang, Yu-Shen Liu, and Matthias Zwicker. Y2seq2seq: Cross-modal representation learning for 3d shape and text by joint reconstruction and prediction of view and word sequences. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 126–133, 2019. 3

[31] Zhizhong Han, Chao Chen, Yu-Shen Liu, and Matthias Zwicker. Shapecaptioner: Generative caption network for 3d shapes by learning a mapping from parts detected in multiple views to sentences. In *Proceedings of the ACM International Conference on Multimedia*, pages 1018–1027, 2020. 3

[32] Zekun Hao, David W Romero, Tsung-Yi Lin, and Ming-Yu Liu. Meshtron: High-fidelity, artist-like 3d mesh generation at scale. *arXiv preprint arXiv:2412.09548*, 2024. 3

[33] Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022. 1

[34] Yining Hong, Haoyu Zhen, Peihao Chen, Shuhong Zheng, Yilun Du, Zhenfang Chen, and Chuang Gan. 3d-llm: Injecting the 3d world into large language models. *Advances in Neural Information Processing Systems*, 36:20482–20494, 2023. 1

[35] Binbin Huang, Zehao Yu, Anpei Chen, Andreas Geiger, and Shenghua Gao. 2d gaussian splatting for geometrically accurate radiance fields. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–11, 2024. 2

[36] Heewoo Jun and Alex Nichol. Shap-e: Generating conditional 3d implicit functions. *arXiv preprint arXiv:2305.02463*, 2023. 2

[37] Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. 2

[38] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 3

[39] Dongting Li, Chenchong Tang, and Han Liu. Audio-llm: Activating the capabilities of large language models to comprehend audio data. In *International Symposium on Neural Networks*, pages 133–142. Springer, 2024. 2

[40] Jinhua Liang, Xubo Liu, Wenwu Wang, Mark D Plumbley, Huy Phan, and Emmanouil Benetos. Acoustic prompt tuning: Empowering large language models with audition capabilities. *IEEE Transactions on Audio, Speech and Language Processing*, 2025. 2

[41] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text Summarization Branches out*, pages 74–81, 2004. 6

[42] Aixin Liu, Bei Feng, Bin Wang, Bingxuan Wang, Bo Liu, Chenggang Zhao, Chengqi Dengr, Chong Ruan, Damai Dai, Daya Guo, et al. Deepseek-v2: A strong, economical, and efficient mixture-of-experts language model. *arXiv preprint arXiv:2405.04434*, 2024. 2

[43] Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, et al. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*, 2024. 2

[44] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. *Advances in Neural Information Processing Systems*, 36:34892–34916, 2023. 1, 2

[45] Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306, 2024. 2

[46] Shansong Liu, Atin Sakkeer Hussain, Chenshuo Sun, and Ying Shan. Music understanding llama: Advancing text-to-music generation with question answering and captioning. In *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 286–290. IEEE, 2024. 1

[47] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 3

[48] Norman Müller, Yawar Siddiqui, Lorenzo Porzi, Samuel Rota Bulo, Peter Kontschieder, and Matthias Nießner. Diffrf: Rendering-guided 3d radiance field diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4328–4338, 2023. 2

[49] Charlie Nash, Yaroslav Ganin, SM Ali Eslami, and Peter Battaglia. Polygen: An autoregressive generative model of 3d meshes. In *International Conference on Machine Learning*, pages 7220–7229. PMLR, 2020. 3, 5, 6, 8

[50] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35: 27730–27744, 2022. 1, 2

[51] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, 2002. 6

[52] Jeong Joon Park, Peter Florence, Julian Straub, Richard Newcombe, and Steven Lovegrove. Deepsdf: Learning continuous signed distance functions for shape representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 165–174, 2019. 2

[53] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019. 1, 2

[54] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 6

[55] Christoph Schuhmann. Improved aesthetic predictor. https://github.com/christophschuhmann/improved-aesthetic-predictor, 2022. 4, 5

[56] Yu Shu, Siwei Dong, Guangyao Chen, Wenhao Huang, Ruihua Zhang, Daochen Shi, Qiqi Xiang, and Yemin Shi. Llasm: Large language and speech model. *arXiv preprint arXiv:2308.15930*, 2023. 1

[57] Yawar Siddiqui, Antonio Alliegro, Alexey Artemov, Tatiana Tommasi, Daniele Sirigatti, Vladislav Rosov, Angela Dai, and Matthias Nießner. Meshgpt: Generating triangle meshes with decoder-only transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 19615–19625, 2024. 3, 6

[58] Yu Sun, Shuohuan Wang, Shikun Feng, Siyu Ding, Chao Pang, Junyuan Shang, Jiaxiang Liu, Xuyi Chen, Yanbin Zhao, Yuxiang Lu, et al. Ernie 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation. *arXiv preprint arXiv:2107.02137*, 2021. 1

[59] Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*, 2023.

[60] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023. 2

[61] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023. 1, 2

[62] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4566–4575, 2015. 6

[63] Zhengyi Wang, Jonathan Lorraine, Yikai Wang, Hang Su, Jun Zhu, Sanja Fidler, and Xiaohui Zeng. Llama-mesh: Unifying 3d mesh generation with language models. *arXiv preprint arXiv:2411.09595*, 2024. 2, 3, 6, 8

[64] Haohan Weng, Yikai Wang, Tong Zhang, CL Chen, and Jun Zhu. Pivotmesh: Generic 3d mesh generation via pivot vertices guidance. *arXiv preprint arXiv:2405.16890*, 2024. 3

[65] Haohan Weng, Zibo Zhao, Biwen Lei, Xianghui Yang, Jian Liu, Zeqiang Lai, Zhuo Chen, Yuhong Liu, Jie Jiang, Chunchao Guo, et al. Scaling mesh generation via compressive tokenization. *arXiv preprint arXiv:2411.07025*, 2024. 3

[66] BigScience Workshop, Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, et al. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*, 2022. 1

[67] Runsen Xu, Xiaolong Wang, Tai Wang, Yilun Chen, Jiangmiao Pang, and Dahua Lin. Pointllm: Empowering large language models to understand point clouds. In *European Conference on Computer Vision*, pages 131–147. Springer, 2024. 1

[68] Yunhan Yang, Yukun Huang, Yuan-Chen Guo, Liangjun Lu, Xiaoyang Wu, Edmund Y Lam, Yan-Pei Cao, and Xihui Liu. Sampart3d: Segment any part in 3d objects. *arXiv preprint arXiv:2411.07184*, 2024. 2, 4, 5

[69] Fukun Yin, Xin Chen, Chi Zhang, Biao Jiang, Zibo Zhao, Jiayuan Fan, Gang Yu, Taihao Li, and Tao Chen. Shapegpt: 3d shape generation with a unified multi-modal language model. *arXiv preprint arXiv:2311.17618*, 2023. 1, 3

[70] Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. Speechgpt: Empowering large language models with intrinsic cross-modal conversational abilities. *arXiv preprint arXiv:2305.11000*, 2023. 2

[71] Hang Zhang, Xin Li, and Lidong Bing. Video-llama: An instruction-tuned audio-visual language model for video understanding. *arXiv preprint arXiv:2306.02858*, 2023. 1, 2